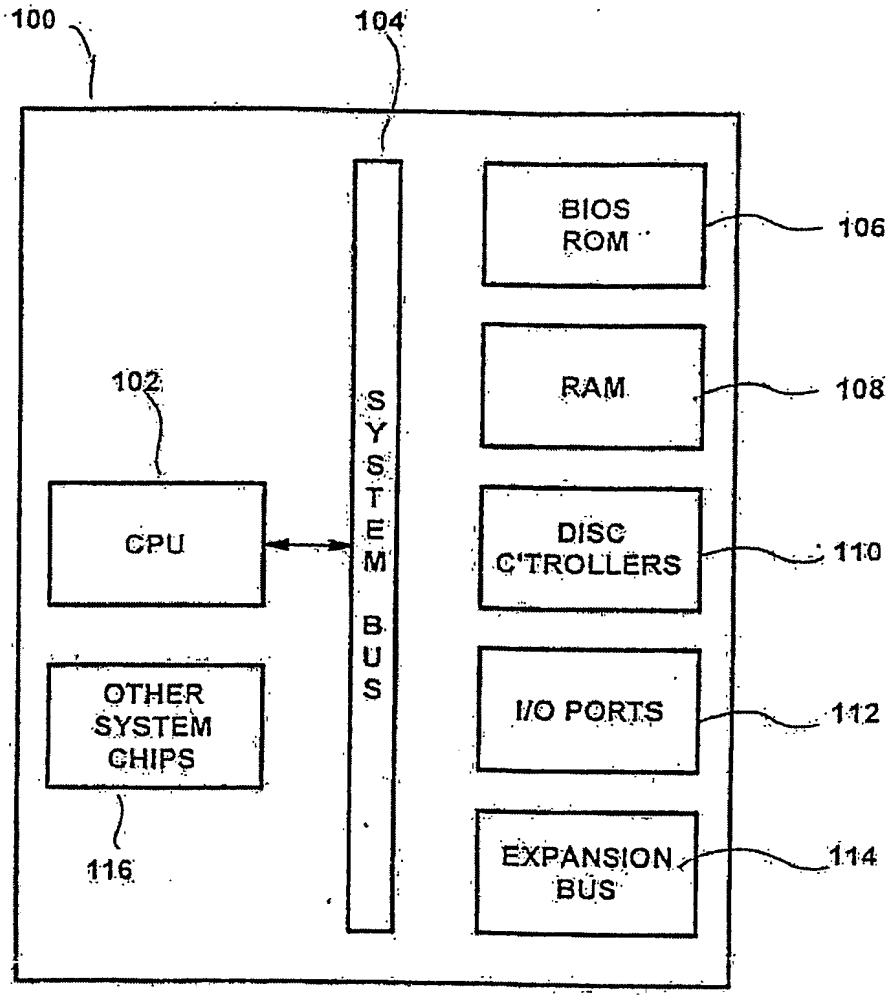

100
104
BIOS ROM
106
RAM
108
102
SYSTEM BUS
CPU
DISC C'TROLLERS
110
OTHER SYSTEM CHIPS
I/O PORTS
112
116
EXPANSION BUS
114

FIG. 1

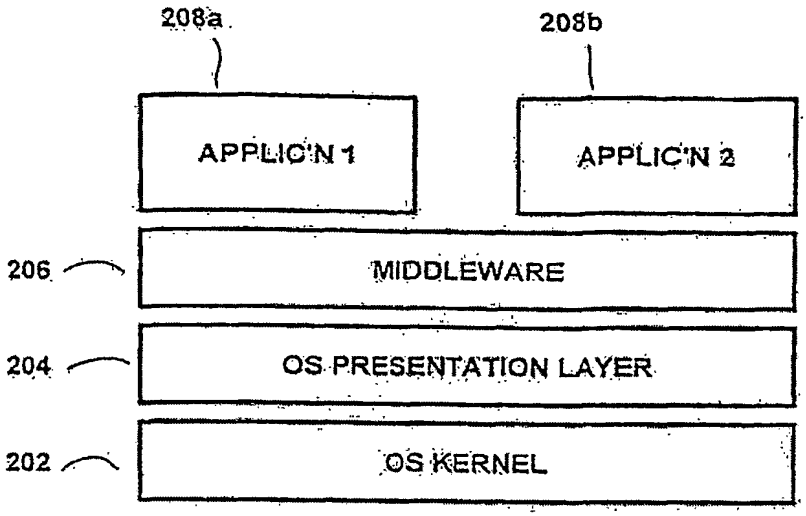

208a
208b
APPLIC'N 1
APPLIC'N 2
206
MIDDLEWARE
204
OS PRESENTATION LAYER
202
OS KERNEL

FIG. 2a

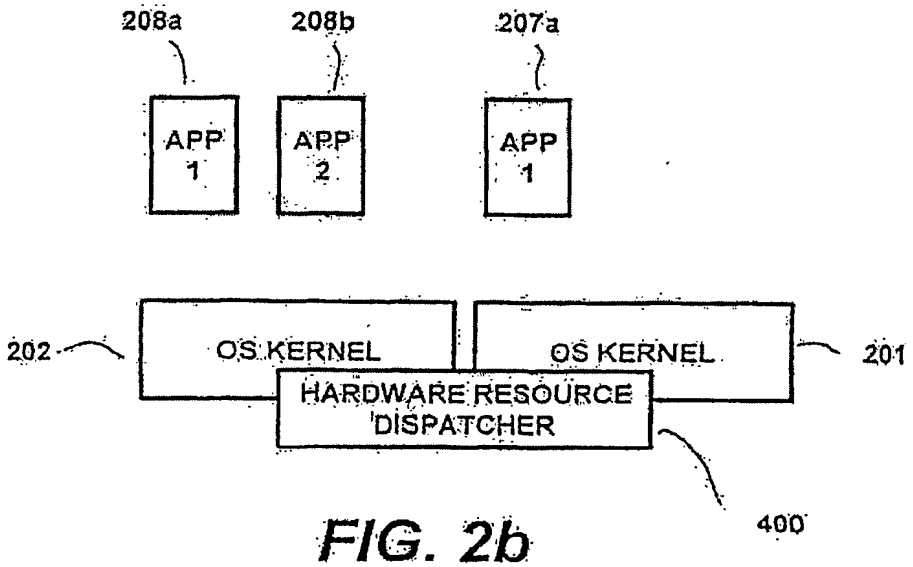
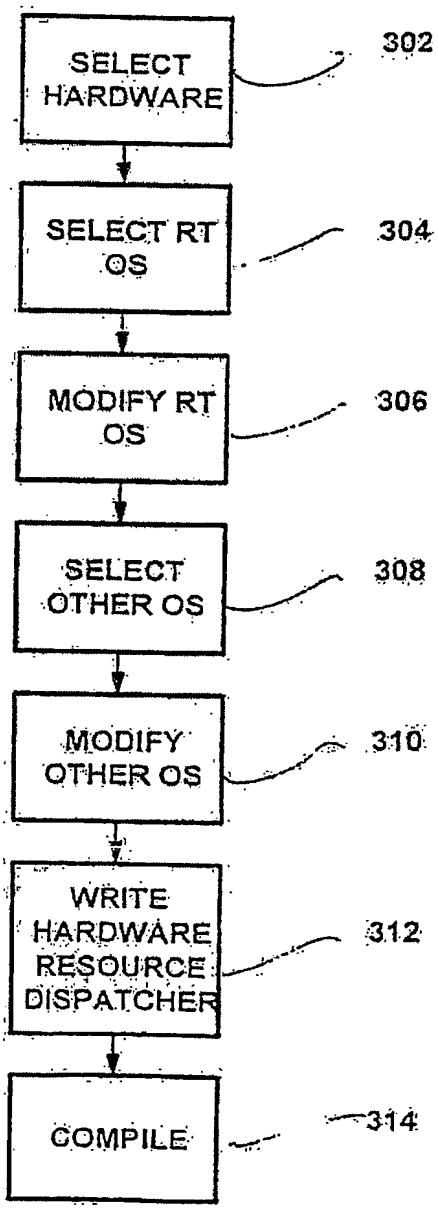

208a
208b
207a
APP 1
APP 2
APP 1
202
OS KERNEL
OS KERNEL
201
HARDWARE RESOURCE DISPATCHER
400

FIG. 2b

FIG. 3


SELECT HARDWARE
302
SELECT RT OS
304
MODIFY RT OS
306
SELECT OTHER OS
308
MODIFY OTHER OS
310
WRITE HARDWARE RESOURCE DISPATCHER
312
COMPILE
314

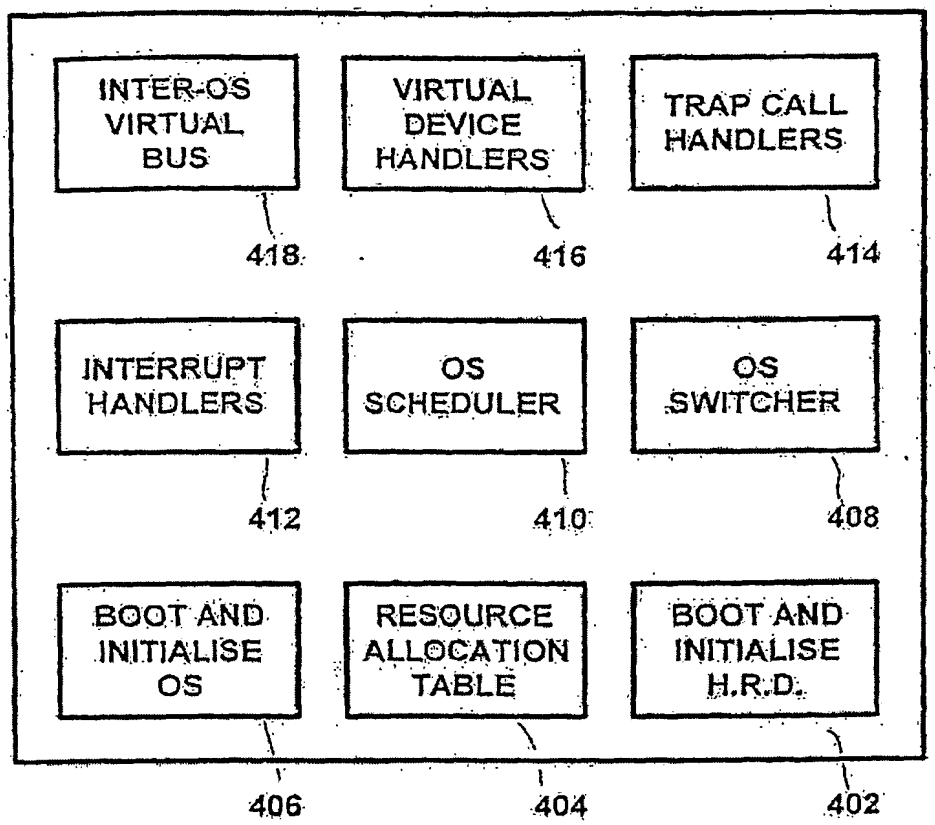

INTER-OS VIRTUAL BUS
VIRTUAL DEVICE HANDLERS
TRAP CALL HANDLERS
418
416
414
INTERRUPT HANDLERS
OS SCHEDULER
OS SWITCHER
412
410
408
BOOT AND INITIALISE OS
RESOURCE ALLOCATION TABLE
BOOT AND INITIALISE H.R.D.
406
404
402

FIG. 4

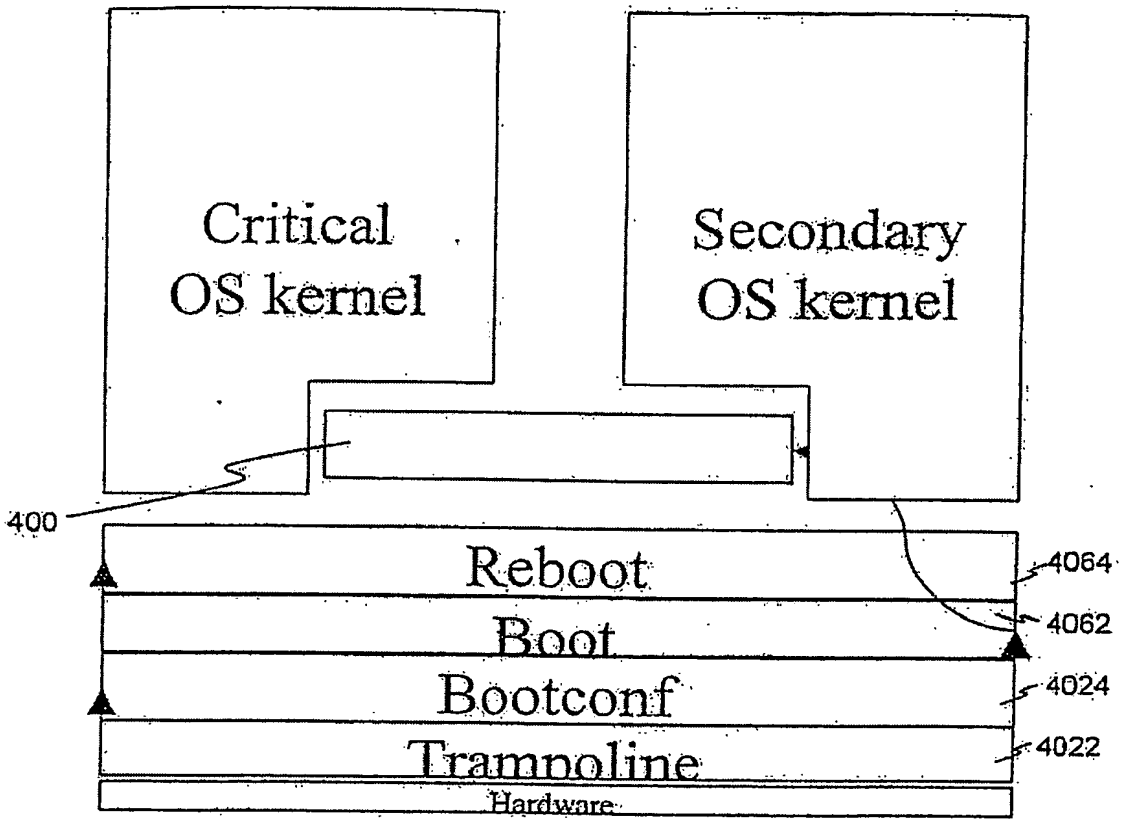

Critical
OS kernel
Secondary
OS kernel
400
Reboot
Boot
Bootconf
Tramnoline
Hardware
4064
4062
4024
4022

FIG. 5

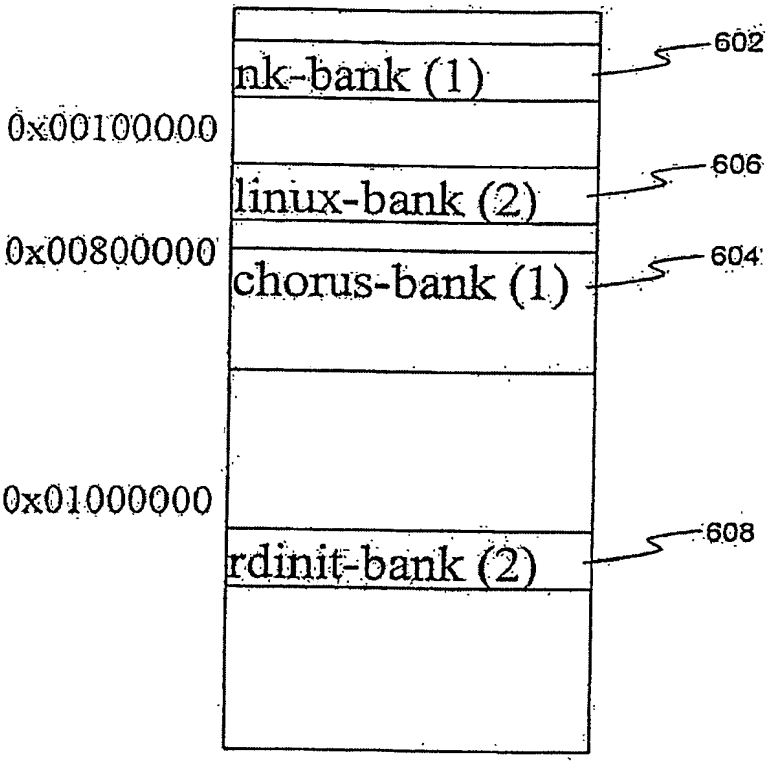

nk-bank (1)
602
0x00100000
linux-bank (2)
606
0x00800000
chorus-bank (1)
604
0x01000000
rdinit-bank (2)
608

FIG. 6

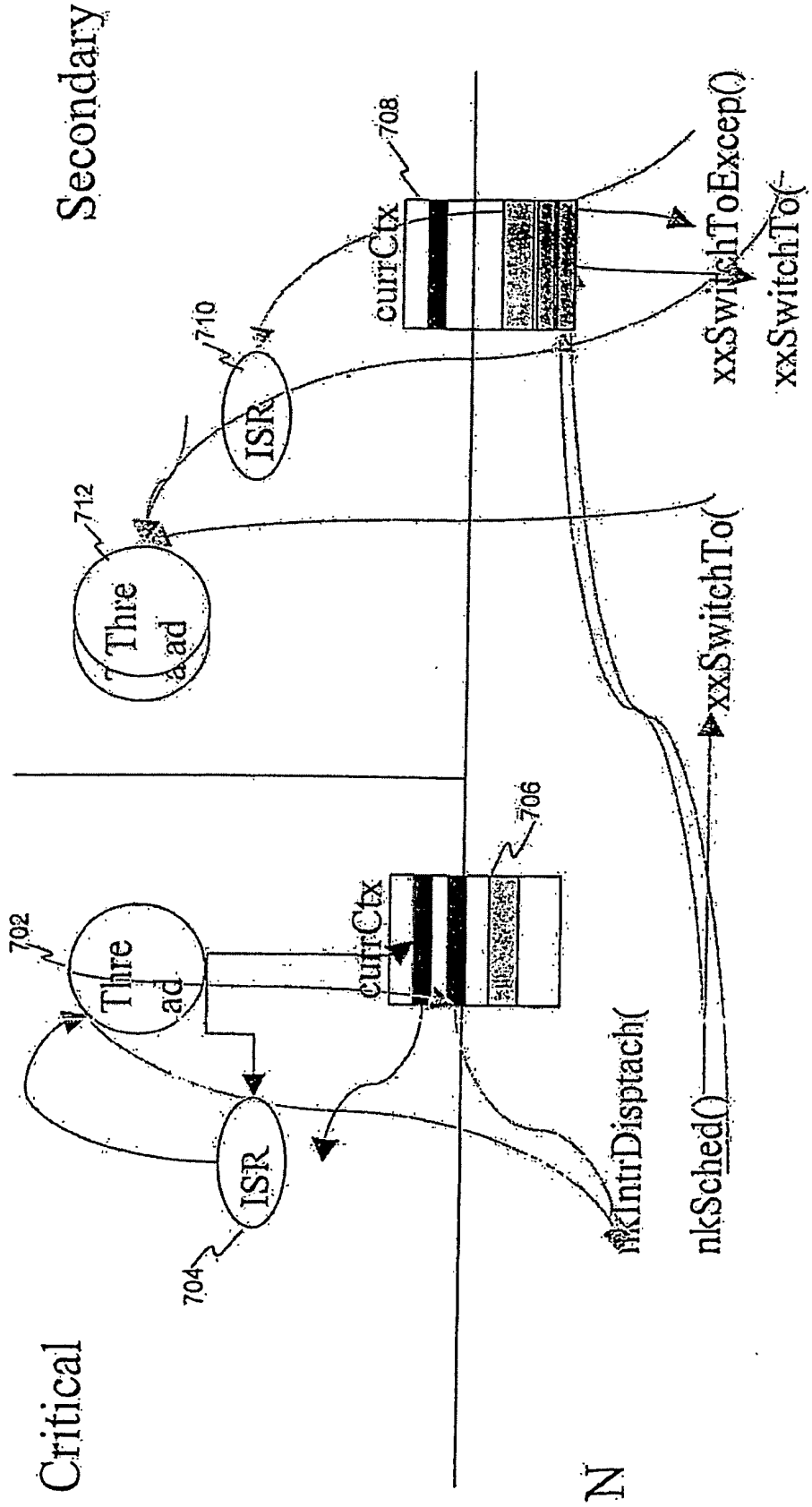

Secondary
Critical
N
Thre ad
ISR
currCtx
xxSwitchToExcep()
xxSwitchTo(
nkSched()
nkIntrDisptach(
702
704
706
708
710
712

FIG. 7

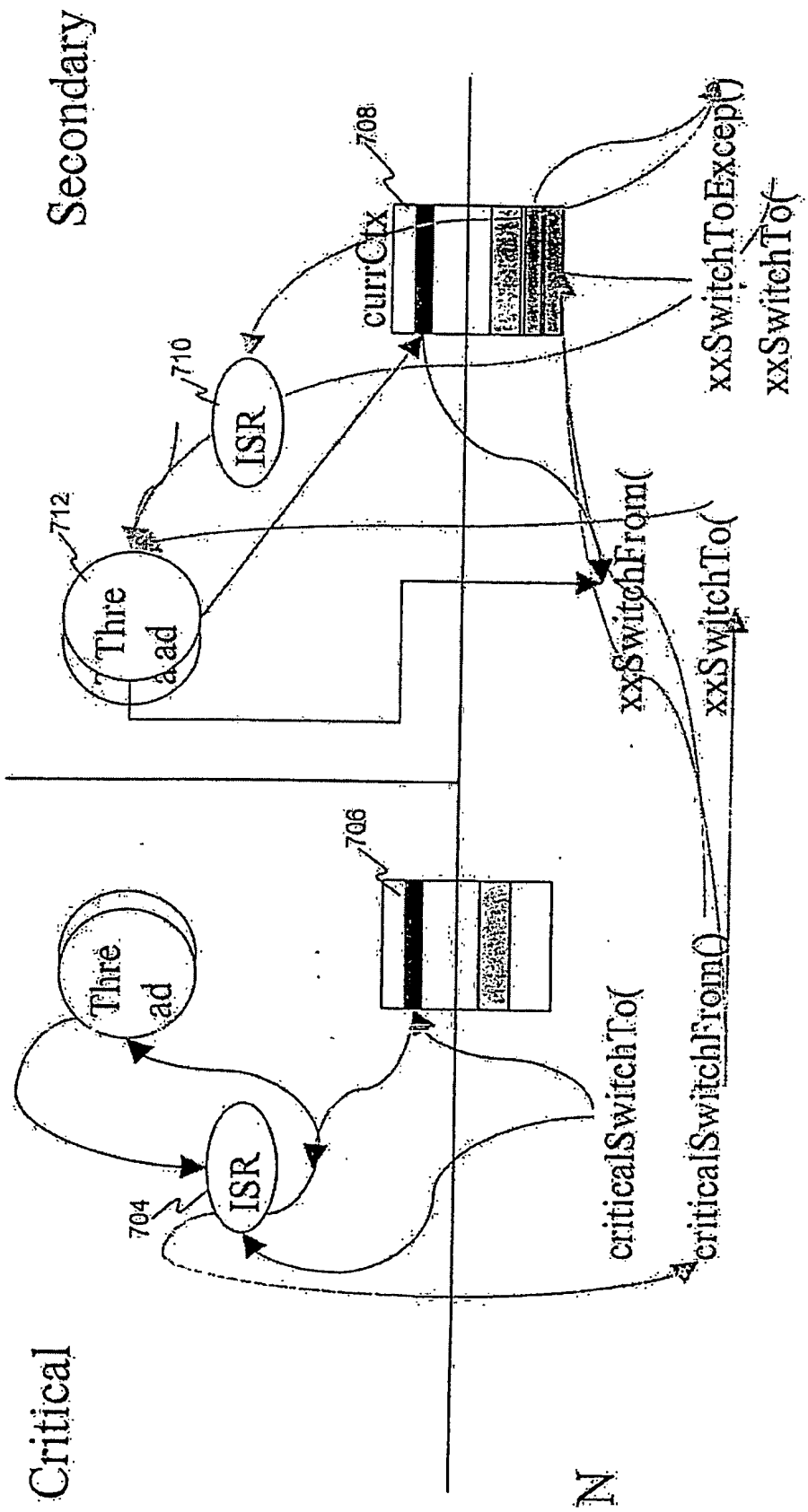

Secondary
Critical
N
Thre ad
ISR
704
706
708
710
712
currCtx
criticalSwitchTo(
criticalSwitchFrom()
xxSwitchFrom(
xxSwitchTo(
xxSwitchToExcep()
xxSwitchTo(

FIG. 8

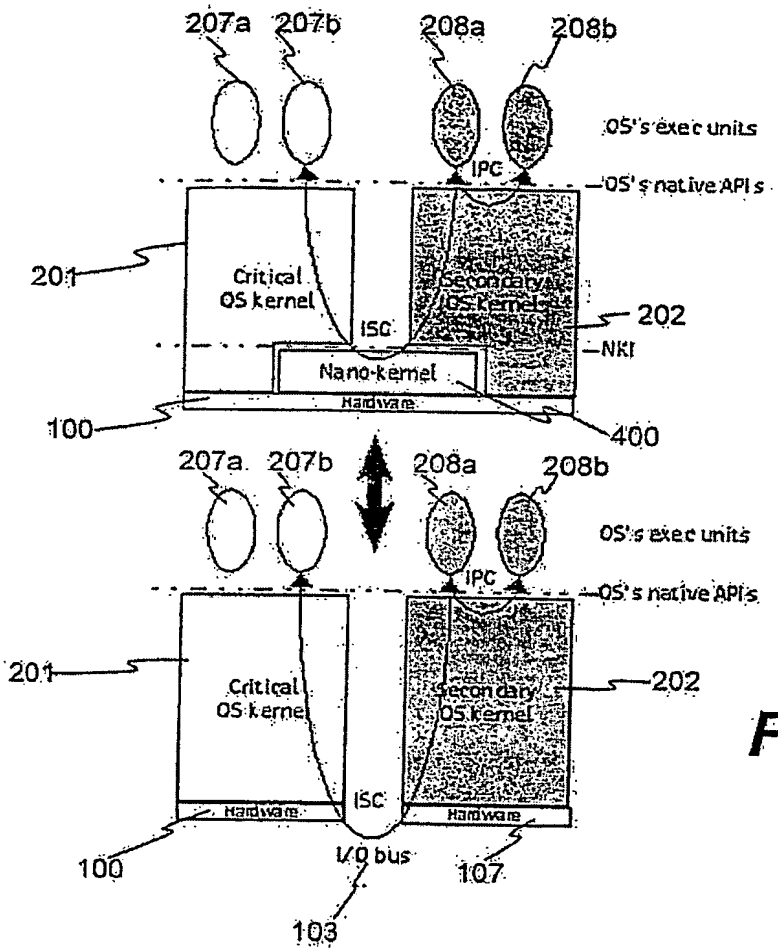

207a
207b
208a
208b
OS's exec units
IPC
OS's native API s
201
Critical OS kernel
ISC
Secondary OS kernel
202
NKI
Nano-kernel
Hardware
100
400

FIG. 9a

207a
207b
208a
208b
OS's exec units
IPC
OS's native API s
201
Critical OS kernel
Secondary OS kernel
202
ISC
Hardware
Hardware
100
I/O bus
107
103

FIG. 9b

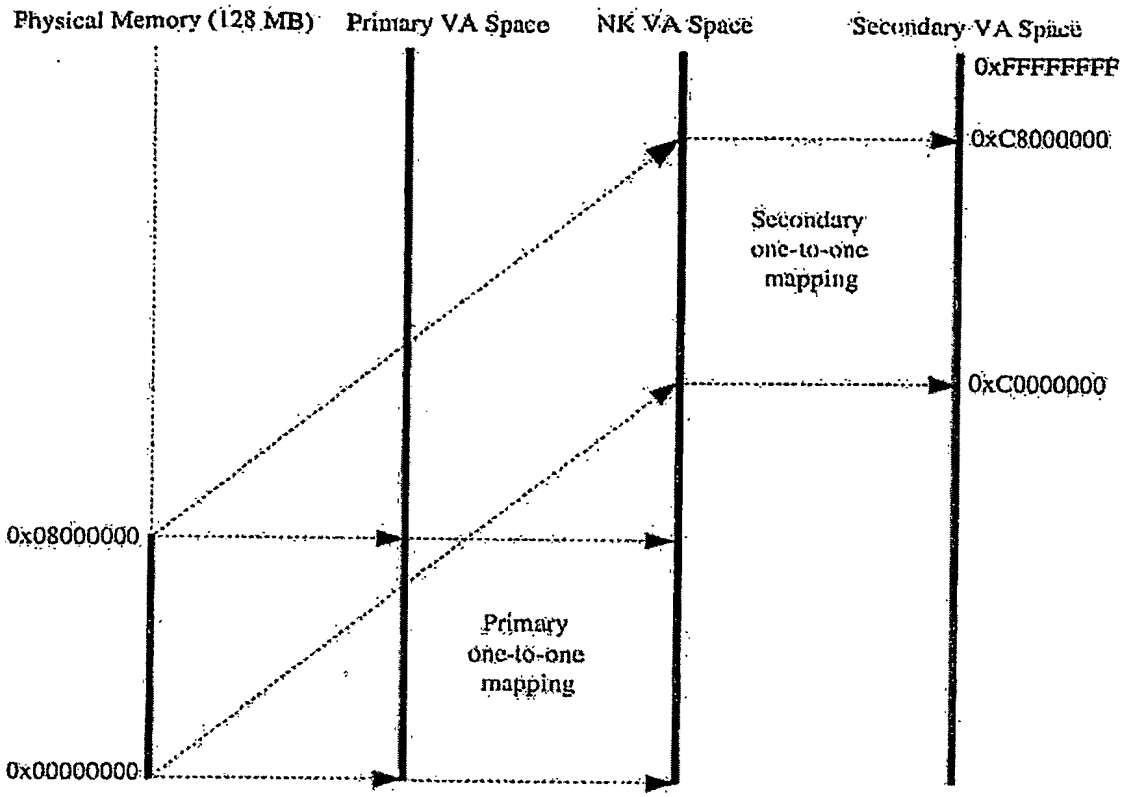


Fig. 10 Virtual Address Spaces

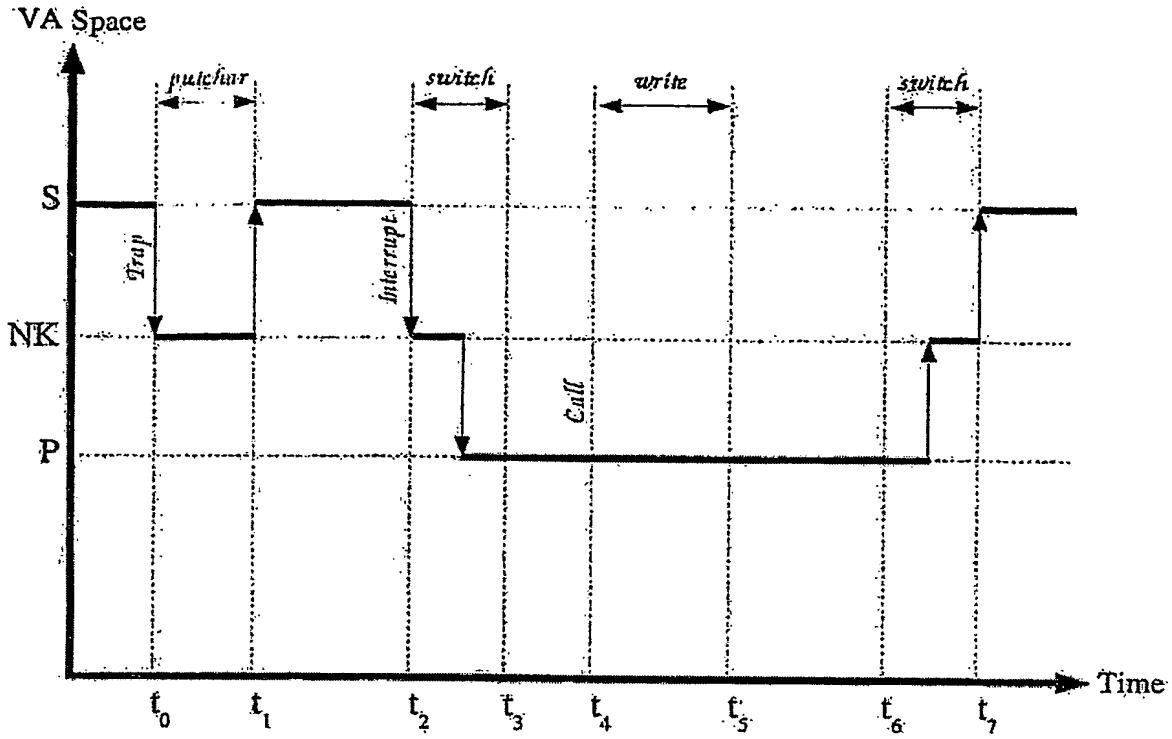


Fig. 11 Virtual Address Space Switching

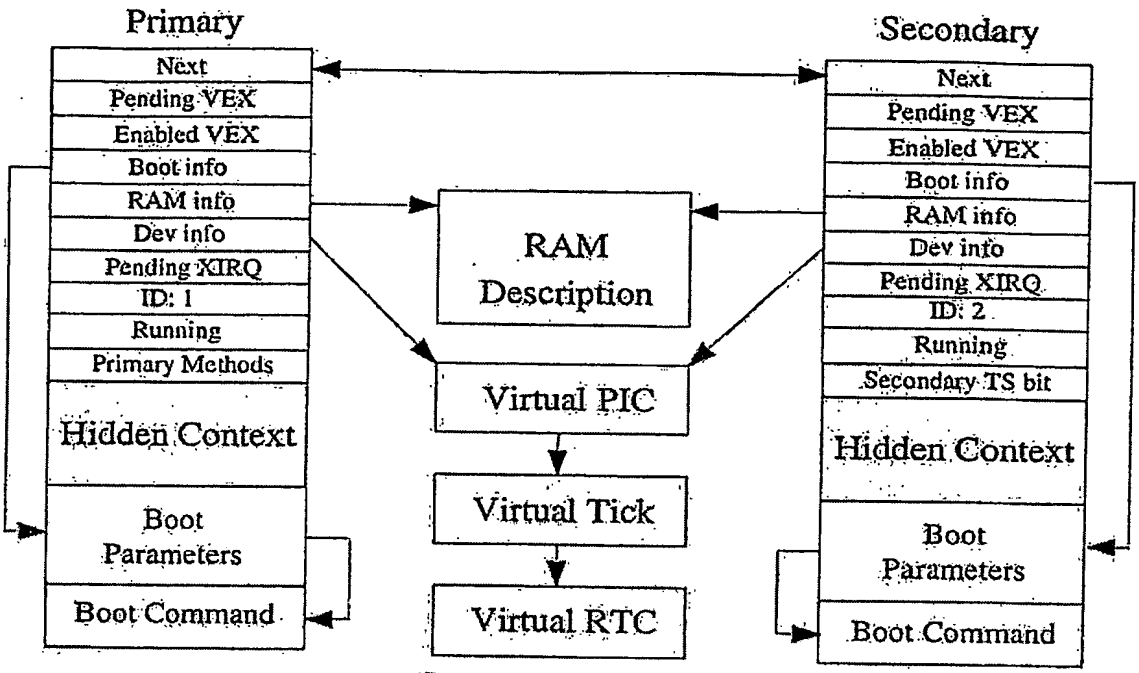


Fig. 12 Visible Kernel Context

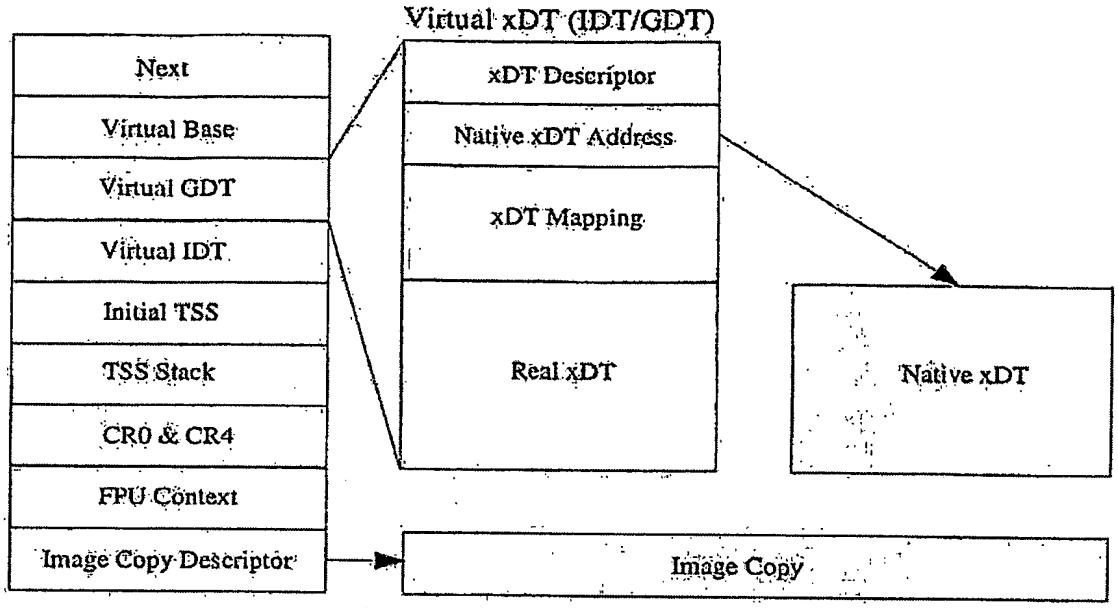


Fig. 13 Hidden Kernel Context

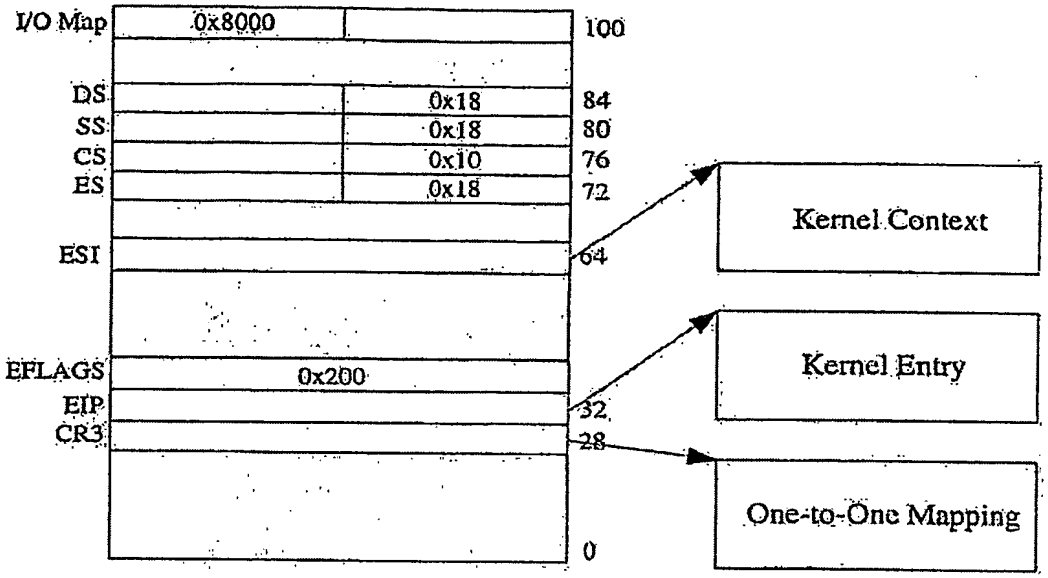


Fig. 14 Initial TSS

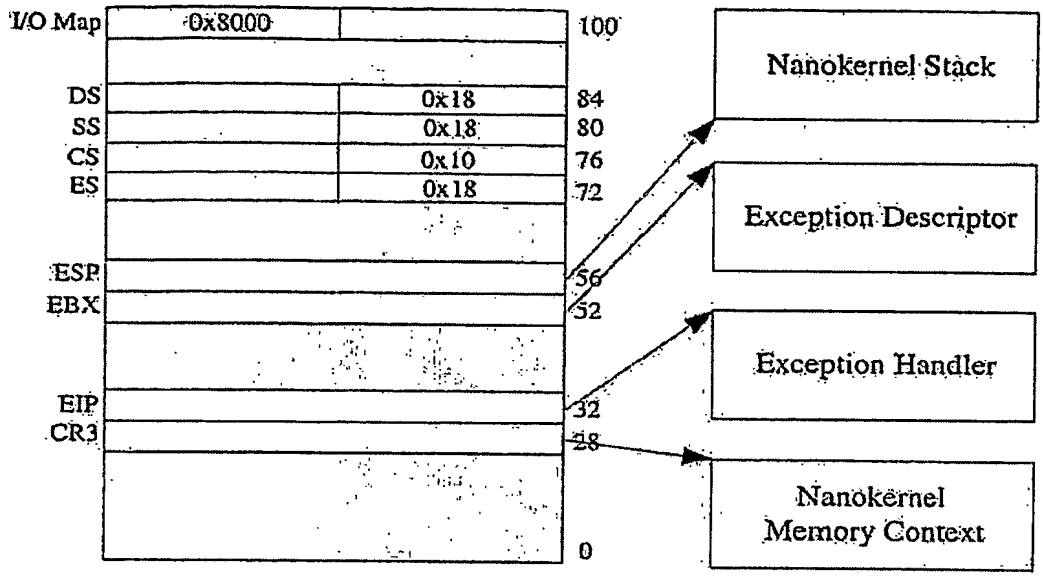


Fig. 15 Nanokernel TSS

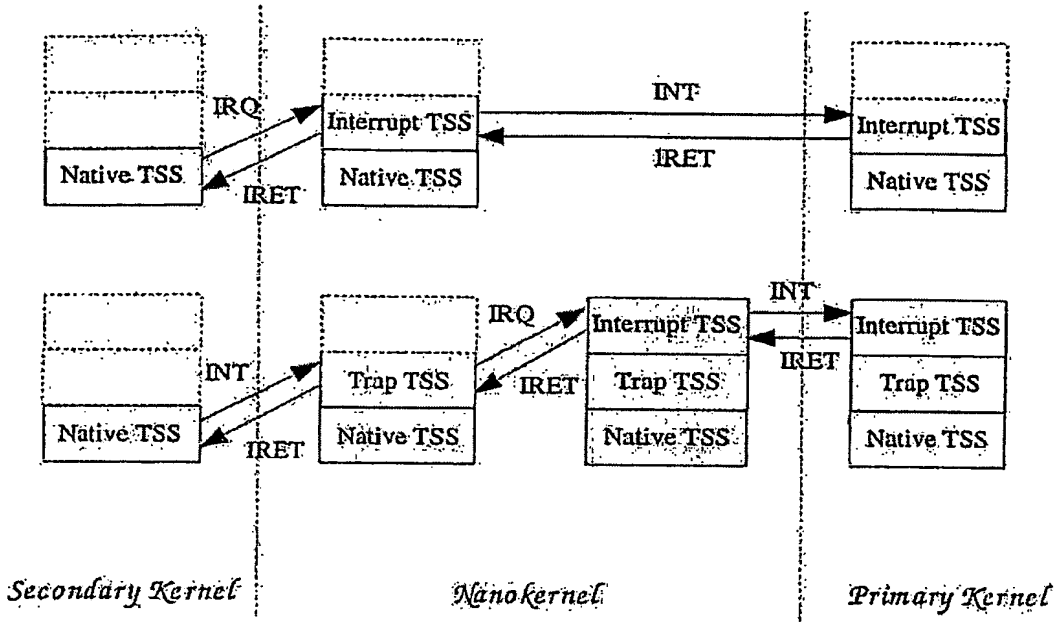


Fig. 16 Nanokernel TSS Stack

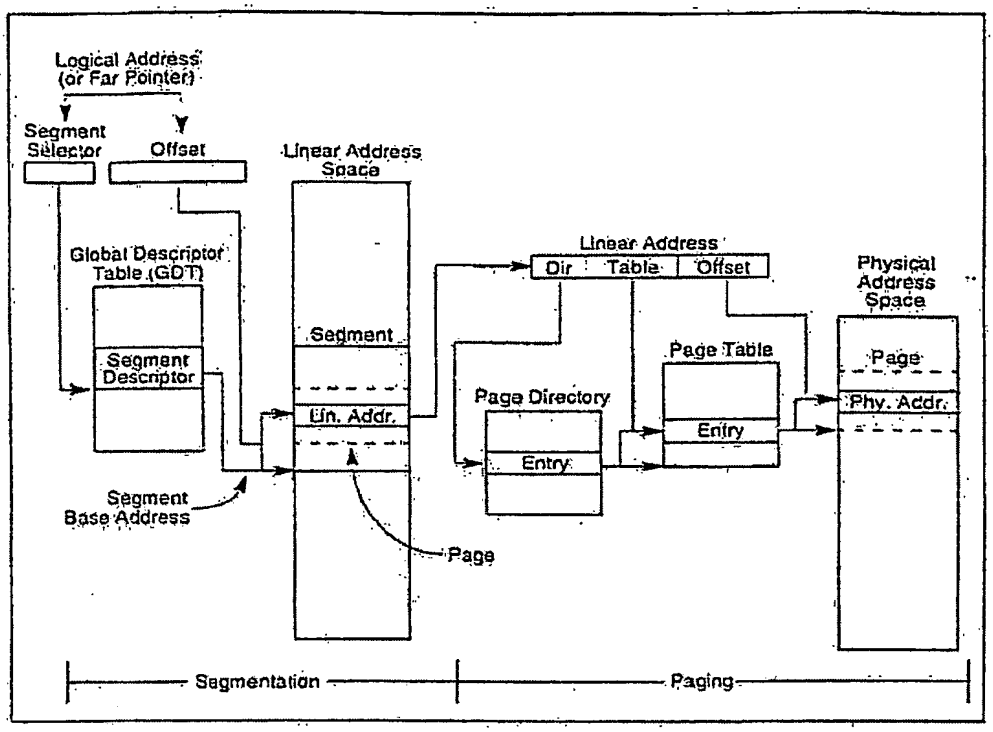

Logical Address
(or Far Pointer)
Segment
Selector
Offset
Linear Address
Space
Global Descriptor
Table (GDT)
Segment
Descriptor
Segment
Lin. Addr.
Segment
Base Address
Page
Linear Address
Dir
Table
Offset
Page Directory
Entry
Page Table
Entry
Physical
Address
Space
Page
Phy. Addr.
Segmentation
Paging

FIG. 17

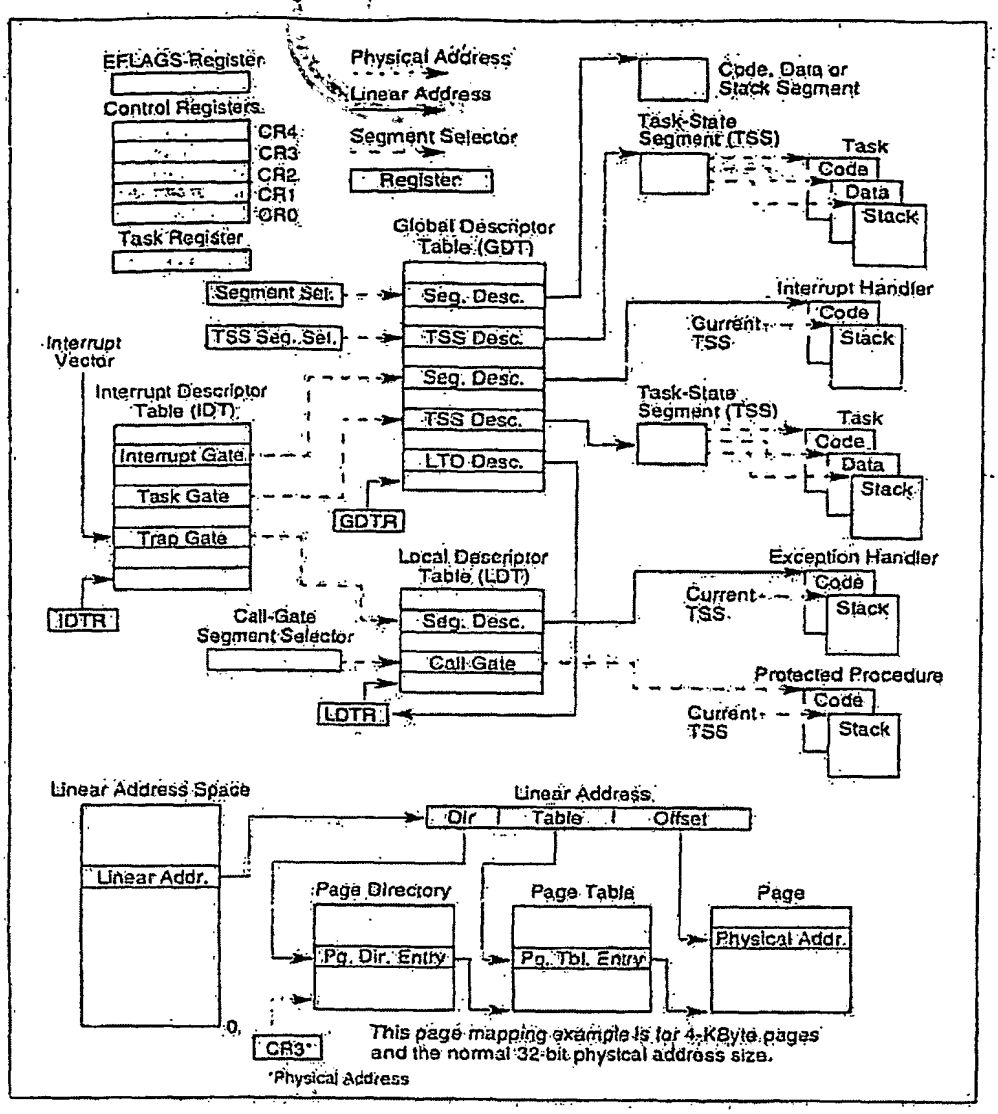

EFLAGS Register
Physical Address
Linear Address
Control Registers
CR4
CR3
CR2
CR1
CR0
Segment Selector
Register
Task Register
Global Descriptor Table (GDT)
Code, Data or Stack Segment
Task-State Segment (TSS)
Task
Code
Data
Stack
Segment Sel.
Seg. Desc.
TSS Seg. Sel.
TSS Desc.
Seg. Desc.
TSS Desc.
LTO Desc.
Interrupt Handler
Code
Stack
Current TSS
Interrupt Vector
Interrupt Descriptor Table (IDT)
Interrupt Gate
Task Gate
Trap Gate
GDTR
IDTR
Local Descriptor Table (LDT)
Call-Gate Segment Selector
Seg. Desc.
Call Gate
LDTR
Exception Handler
Protected Procedure
Linear Address Space
Linear Addr.
Linear Address
Dir
Table
Offset
Page Directory
Page Table
Page
Pg. Dir. Entry
Pg. Tbl. Entry
Physical Addr.
0
CR3*
*Physical Address
This page mapping example is for 4-KByte pages and the normal 32-bit physical address size.

FIG. 18